

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claim 1 (currently amended): An apparatus comprising:

at least one processor;

a memory coupled to the at least one processor, wherein the memory stores non-object oriented data; and

mapping software residing in memory, wherein the at least one processor executes the mapping software to map an object to an image in memory occupied by the non-object oriented data located in the memory using zero size object mapping, wherein the zero size object mapping does not require memory in addition to a portion of the memory storing the non-object oriented data and ~~wherein the object oriented data inherits the non-object oriented data~~wherein the mapping software provides bidirectional access between the object and the non-object oriented data.

Claims 2-3 (canceled)

Claim 4 (currently amended): A method for retrieving non-object oriented data from within an object oriented model, the method comprising the steps of:

loading memory with non-object oriented data;

mapping an object oriented model to an image in memory occupied by the non-object oriented data without requiring additional memory space; and

retrieving a non-object oriented data element from the memory in the object oriented model based on the mapping, wherein the object oriented data inherits the non-object oriented data; and

providing bi-directional access, wherein the bi-directional access permits retrieval of the object oriented data based on the non-object oriented data.

Claim 5 (canceled)

Claim 6 (currently amended): The method of claim 6- 4 wherein the step of mapping further comprising:

deriving a class from the non-object oriented data.

Claim 7 (previously presented): The method of claim 6, wherein the step of mapping further comprising:

instantiating an instance of the class based on static casting.

Claim 8 (canceled)

Claim 9 (previously presented): The method of claim 4, wherein the step of mapping further comprising:

accessing the non-object oriented data using the object oriented model.

Claim 10 (previously presented): The method of claim 4, wherein the step of retrieving occurs with zero size memory.

Claims 11-23 (canceled)

Claim 24 (currently amended): A method for providing a bi-directional access between non-object oriented data and object oriented data comprising:

mapping a data object representing object oriented data to an image in memory occupied by non-object oriented data represented by legacy C-structure data;

creating a child class based on inheriting the legacy C-structure data by the data object;

accessing the child class by static casting the non-object oriented data represented by the legacy C-structure data with the object oriented data; and

providing bi-directional access, wherein the bi-directional access permits retrieval of the object oriented data based on the non-object oriented data.

Claim 25 (previously presented): The method of claim 24, wherein using zero size mapping is used to map the data object representing object oriented data to the image in memory occupied by the non-object oriented data.

Claim 26 (previously presented): The method of claim 25, further comprising:

deriving object oriented classes from the legacy C-structures at compile time.

Claim 27 (previously presented): The method of claim 26, further comprising:

loading memory with non-object oriented data into the derived object oriented classes.

Claim 28 (previously presented): The method of claim 24, wherein the non-object oriented data and the object oriented data are associated with very large scale integrated circuits.